OIPE IAP17 JAN 16 2007 PATENT & TRADEMARK OFFICE

AF / 2628

DOCKET NO.: 05-03-014
CLIENT NO.: UGSC01-05031
Customer No.: 45113

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of	:	JIANBING HUANG ET AL.
U.S. Serial No.	:	10/731,281
Filed	:	December 9, 2003
For	:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.	:	2628
Examiner	:	Kimbinh T. Nguyen

MAIL STOP APPEAL BRIEF -PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

CERTIFICATE OF MAILING BY FIRST CLASS MAIL

The undersigned hereby certifies that the following documents:

1. Appeal Brief;
2. Fee Transmittal FY 2007 (in duplicate);
3. Check in the amount of $500.00 for Appeal Brief filing fee; and
4. Postcard receipt

relating to the above application, were deposited as "First Class Mail" with the United States Postal Service, addressed to MAIL STOP APPEAL BRIEF -PATENTS, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on January 9, 2007.

Date: 1/9/07

Mailer

Date: 1/9/7

Matthew S. Anderson
Reg. No. 39,093

P.O. Drawer 800889
Dallas, Texas 75380
Phone: (972) 628-3600
Fax: (972) 628-3616
E-mail: *manderson@munckbutrus.com*

OIPE IAP17 JAN 16 2007 PATENT & TRADEMARK OFFICE

PTO/SB/17 (12-04)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995 no persons are required to respond to a collection of information unless it displays a valid OMB control number

Effective on 12/08/2004.
Fees pursuant to the Consolidated Appropriations Act, 2005 (H.R. 4818).

FEE TRANSMITTAL
For FY 2007

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT ($) 500.00

Complete if Known	
Application Number	10/731,281
Filing Date	December 9, 2003
First Named Inventor	Jianbing Huang
Examiner Name	Kimbinh T. Nguyen
Art Unit	2628
Attorney Docket No.	05-03-014

METHOD OF PAYMENT (check all that apply)

[X] Check ☐ Credit Card ☐ Money Order ☐ None ☐ Other (please identify): ____

[X] Deposit Account Deposit Account Number: 50-0208 Deposit Account Name: Munck Butrus, P.C.

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

☐ Charge fee(s) indicated below ☐ Charge fee(s) indicated below, **except for the filing fee**

[X] Charge any additional fee(s) or underpayments of fee(s) under 37 CFR 1.16 and 1.17 [X] Credit any overpayments

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

FEE CALCULATION

1. BASIC FILING, SEARCH, AND EXAMINATION FEES

Application Type	FILING FEES Fee ($)	FILING FEES Small Entity Fee ($)	SEARCH FEES Fee ($)	SEARCH FEES Small Entity Fee ($)	EXAMINATION FEES Fee ($)	EXAMINATION FEES Small Entity Fee ($)	Fees Paid ($)
Utility	300	150	500	250	200	100	
Design	200	100	100	50	130	65	
Plant	200	100	300	150	160	80	
Reissue	300	150	500	250	600	300	
Provisional	200	100	0	0	0	0	

2. EXCESS CLAIM FEES

Fee Description	Fee ($)	Small Entity Fee ($)
Each claim over 20 or, for Reissues, each claim over 20 and more than in the original patent	50	25
Each independent claim over 3 or, for Reissues, each independent claim more than in the original patent	200	100
Multiple dependent claims	360	180

Total Claims ____ - 20 or HP = ____ x ____ = ____ (Extra Claims, Fee ($), Fee Paid ($))
HP = highest number of total claims paid for, if greater than 20

Multiple Dependent Claims: Fee ($) ____ Fee Paid ($) ____

Indep. Claims ____ - 3 or HP = ____ x ____ = ____ (Extra Claims, Fee ($), Fee Paid ($))
HP = highest number of independent claims paid for, if greater than 3

3. APPLICATION SIZE FEE

If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).

Total Sheets ____ - 100 = Extra Sheets ____ / 50 = Number of each additional 50 or fraction thereof ____ (round up to a whole number) x Fee ($) ____ = Fee Paid ($) ____

4. OTHER FEE(S) Fees Paid ($)

Non-English Specification, $130 fee (no small entity discount) ____

Other: Appeal Brief $500.00

SUBMITTED BY				
Signature	[signature]	Registration No. (Attorney/Agent)	39,093	Telephone 972-628-3600
Name (Print/Type)	Matthew S. Anderson			Date January 9, 2007

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

Applicants herewith respectfully submit that the Examiner's decision of August 11, 2006, finally rejecting Claims 1-24 in the present application, should be reversed, in view of the following arguments and authorities. This Brief is submitted on behalf of Appellant for the application identified above. A check is enclosed for the fee for filing a Brief on Appeal. Please charge any additional necessary fees to Deposit Account No. 50-0208.

TABLE OF CONTENTS

APPENDIX A - Claims Appendix

APPENDIX B - Copy of Formal Drawings

APPENDIX C - Evidence Appendix.- No additional evidence was submitted.

APPENDIX D - Related Proceedings Appendix - There are no related proceedings.

TABLE OF AUTHORITIES

Real Party in Interest

The real party in interest, and assignee of this case, is UGS Corp.

Related Appeals or Interferences

To the best knowledge and belief of the undersigned attorney, there are none.

Status of Claims

Claims 1-30 are under final rejection, and are each appealed. Applicant notes that while the Office Action Summary sent with the final Office Action only indicates that claims 1-24 were rejected, a rejection was made to all claims.

Status of Amendments after Final

An after-final Amendment to correct a typographic error in Claim 24 was entered, and is reflected in the claims appendix.

SUMMARY OF CLAIMED SUBJECT MATTER

The following summary refers to disclosed embodiments and their advantages, but does not delimit any of the claimed inventions.

In General

The present application is directed, in general, to graphics processing, and, more specifically, to a system, method, and computer program product that accepts raw polygon geometry and view parameters from a visualization API, sorts the polygons in back-to-front order, and then supplies the sorted triangles to graphics API. *Page 1, lines 4-5, and page 4, lines 2-6.*

Support for Independent Claims

Note that, per 37 CFR §41.37, only each of the independent claims and claims including means-plus-function language are discussed in this section. In the arguments below, however, the dependent claims are also discussed and distinguished from the prior art. The discussion of the claims is for illustrative purposes, and is not intended to effect the scope of the claims.

Independent Claim 1 describes method for graphics processing, including receiving node and view data for a graphic object (1005). The method also includes building a binary-space-partition tree corresponding to the graphic object (1010), the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf . The method also includes sorting shapes at each leaf of the binary-space-partition tree (1015), and outputting the sorted shapes (1020). *Page 11, lines 18-23; page 32, line 22 - page 33, line 5; and Figure 10.*

Independent Claim 8 describes a method for graphics processing, including analyzing shapes in a graphic object (1105). The method also includes creating a root node and a list of additional nodes for a binary-space-partition tree (1110), each node associated with up to a predetermined number of at least one shape. The method also includes performing a partition plane selection for each additional node (1115). The method also includes classifying the shapes at the additional node according to the partition plane selection (1120). The method also includes creating child nodes according to the shape classification (1125). *Page 11, lines 18-23; page 33, lines 6 - 23; and Figure 11.*

Independent Claim 11 describes a data processing system having at least a processor (102) and accessible memory (104, 108), and means for performing steps corresponding to the method of

claim 1. These include means (122) for receiving node and view data for a graphic object (1005). The system also includes means (102) for building a binary-space-partition tree corresponding to the graphic object (1010), the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf. The system also includes means (102) for sorting shapes at each leaf of the binary-space-partition tree (1015), and means (122, 110) for outputting the sorted shapes (1020). Dependent Claim 13 further describes means (104, 108) for caching the shape data, and dependent Claim 14 further describes means (102) for traversing the binary-space-partition tree. *Page 8, lines 25 - page 11, line 4; page 11, lines 18-23; page 15, lines 6-18; page 30, line 34 - page 32, line 21; page 32, line 22 - page 33, line 5; and Figures 1 and 10.*

Independent Claim 18 describes a data processing system having at least a processor (102) and accessible memory (104, 108), and means for performing steps corresponding to the method of claim 18. These include means (102) for analyzing shapes in a graphic object. The system also includes means (102) for creating a root node and a list of additional nodes for a binary-space-partition tree (1110), each node associated with up to a predetermined number of at least one shape. The system also includes means (102) for performing a partition plane selection for each additional node (1115). The method also includes means for classifying the shapes at the additional node according to the partition plane selection (1120). The method also includes means (102) for creating child nodes according to the shape classification (1125). *Page 8, lines 25 - page 11, line 4; page 11, lines 18-23; page 33, lines 6 - 23; and Figures 1 and 11.*

Independent Claim 21 describes a computer program product tangibly embodied in a machine-readable medium, comprising instructions for performing steps corresponding to the method of claim 1. These include instructions for receiving node and view data for a graphic object (1005) and instructions for building a binary-space-partition tree corresponding to the graphic object (1010), the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf. The computer program product further includes instructions for sorting shapes at each leaf of the binary-space-partition tree (1015), and instructions for outputting the sorted shapes (1020). *Page 11, lines 18-23; page 32, line 22 - page 33, line 5; page 35, lines 1-17; and Figure 10.*

Independent Claim 28 describes a computer program product tangibly embodied in a machine-readable medium, comprising instructions for performing steps corresponding to the method of claim 8. These include instructions for creating a root node and a list of additional nodes

for a binary-space-partition tree (1110), each node associated with up to a predetermined number of at least one shape. The computer program product further includes instructions for performing a partition plane selection for each additional node (1115). The computer program product further includes instructions for classifying the shapes at the additional node according to the partition plane selection (1120). The computer program product further includes instructions for creating child nodes according to the shape classification (1125). *Page 11, lines 18-23; page 33, lines 6 - 23; page 35, lines 1-17; and Figure 11.*

Grounds of Rejection to be Reviewed on Appeal

1. Are Claims 1, 2, 4, 5, 8-12, 14, 15, 18-22, 24, 25, and 28-30 obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire *et al.* ("Brokenshire")?

2. Are Claims 3, 6, 7, 13, 16, 17, 23, 26, and 27 obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire *et al.* ("Brokenshire") and further in view of U.S. Patent Application Publication No. 2004/0114794 to Vlasic *et al.* ("Vlasic")?

ARGUMENT

Stated Grounds of Rejection

The rejections outstanding against the Claims are as follows:

Claims 1, 2, 4, 5, 8-12, 14, 15, 18-22, 24, 25, and 28-30 were rejected as obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire et al. ("Brokenshire").

Claims 3, 6, 7, 13, 16, 17, 23, 26, and 27 were rejected as obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire et al. ("Brokenshire") and further in view of U.S. Patent Application Publication No. 2004/0114794 to Vlasic et al. ("Vlasic").

Each claim under each ground of rejection is addressed separately.

Legal Standards

The legal standards for an obviousness[1] rejection are referenced in the footnote below.

---

[1]The Supreme Court has explained how to apply §103:

> Under §103, the scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background, the obviousness or nonobviousness of the subject matter is determined.
>
> *Graham v. John Deere Co.*, 383 U.S. 1, 148 U.S.P.Q. 459, 467 (1966).

Obviousness cannot be inferred from a combination of references without a showing that one of ordinary skill would have been motivated to combine those references:

> When prior art references require selective combination ... to render obvious a subsequent invention, there must be some reason for the combination other than the hindsight gained from the invention itself.... Something in the prior art as a whole must suggest the desirability, and thus the obviousness, of making the combination.
>
> *Uniroyal, Inc. v. Rudkin-Wiley Corp.*, 5 U.S.P.Q.2d 1434, 1438 (Fed.Cir. 1988), *quoting Interconnect Planning Corp. v. Feil*, 227 U.S.P.Q. 543 (Fed.Cir. 1985), *and Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick*, 221 U.S.P.Q. 481 (Fed.Cir. 1984).

Analysis of Examiner's Rejection

The cited references are each briefly discussed in relevant part, and then the rejection of each claim is addressed separately under each ground of rejection.

Korobkin, the primary reference used in the final Office Action, is drawn to a photogrammetry engine for model construction. While Korobkin shares some similarities with the instant application, such as using a binary space partition tree, it does not include several claimed elements and functions, as described in detail below.

Brokenshire is drawn to tightened bounding volumes for BSP-trees, and does not supply key teachings also missing from the primary reference, as described in detail below.

Vlasic is a method and system for interactively rendeing objects with surface light fields and view-dependent opacity. Vlasic does not supply key teachings also missing from the primary reference, as described in detail below.

Ground of Rejection 1: Claims 1, 2, 4, 5, 8-12, 14, 15, 18-22, 24, 25, and 28-30 were rejected as obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire et al. ("Brokenshire").

Claim 1

Claim 1 requires:

> A method for graphics processing, comprising:
>
> receiving node and view data for a graphic object;
>
> building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least

one shape associated with each leaf;

sorting shapes at each leaf of the binary-space-partition tree; and

outputting the sorted shapes.

Examiner Nguyen concedes that Korobkin fails to teach a binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf, as required by claim 1.

Examiner Nguyen instead cites to Brokenshire solely for allegedly teaching at least one shape associated with each leaf. Brokenshire, however, teaches a narrowly directed system for reducing the subspace boundary in a binary space partitioning tree. Examiner Nguyen refers to Brokenshire at col. 6, line 1 - col. 7, line7. For the convenience of the Board, this passage is reproduced in its entirety:

> FIG. 6B shows the BSP tree constructed based on the subspaces in FIG. 6A. The plane A is the root of the tree and its two subplanes B and C are the children of node A. In a similar manner, nodes D and E are children of node B and nodes F and G are children of node C. The position of each subplane is given in the form of the coordinates for the lower left corner and the coordinates of the upper right corner. So the position of plane A is specified by {(0, 0), (26, 14)}. The positions for all the other nodes in the tree are given on the figure.
>
> FIG. 7A shows the same two dimensional space containing an image of a pickup truck as in FIG. 6A. The same initial subplanes, A through G, are formed by recursively subdividing subspaces. It is discovered that subplane D has a lot of "dead space"; this means the plane has a large area that does not overlap the image of the pickup truck. Subplane D can be reduced in size, as shown in FIG. 7A, and still cover the image of the pickup truck. In a similar manner, subplane F can be reduced in size substantially and still cover the image of the pickup truck.
>
> FIG. 7B shows the BSP tree constructed based on the subspaces in FIG. 7A in accordance with a preferred embodiment of the invention. The tree has the same structure as that shown in FIG. 6B, but the sizes of the subplanes D and F have been reduced. In particular, the size of D is reduced from a 13x7 unit rectangle to a 5.times.4 rectanglè

> positioned from (8, 7) to (13, 11). The units can be any type of units, however for purposed of the present description, the units will be omitted for simplicity. The size of F is reduced from a 13.times.7 rectangle to a 2.times.4 rectangle positioned from (13, 7) to (15, 11).
>
> With reference now to FIG. 8, a flowchart illustrating an exemplary process and program flow for determining which subspaces should have their bounds tightened and the values of the tightened bounds is depicted in accordance with the present invention. This process may be performed on a server, such as, for example, server 104 in FIG. 1, with the results transmitted to a client, such as, for example, client 108 in FIG. 1, to be displayed on the client. First, a space, or root node, within a binary space partitioning tree is created for an object for which an image is to be rendered (step 802). The space should include the entire bounds of the object that is to be rendered and displayed. Next, the space partitioned into a plurality of first level subspaces to create child nodes or child subspaces (step 804). Each first level subspace may then partitioned into a plurality of second level subspaces (step 806). However, each subspace does not necessarily need to be subdivided further and some subspaces may be subdivided to more levels than others as determined by the requirements of the object that is to be rendered.
>
> The rendering program then continues partitioning each subspace, if desired, such that at each level, each subspace is partitioned into further subspaces until a desired number of levels of subspaces has been created. (step 808). For three dimensions, each subspace may be a rectangular solid or some other polyhedron for which a space may be completely mapped without gaps that do not fall within any subspace. However, rectangular solids provide the simplest and easiest to use subspace. In two dimensions, the subspace may be rectangular as illustrated in FIGS. 6A and 7A, or some other polygon, again with the caveat that subspaces must be of a shape that the entire plane could be mapped without any gaps in the space that fall within an area not contained within any of the subspaces. However, it should be noted, that the entire space may not be mapped by the sum of the reduced subspaces as described herein, since such area may contain no information about the object whose image is to be rendered and displayed.

As can be seen, although Brokenshire teaches subplanes within planes, Brokenshire fails to teach or disclose a binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf, as required by independent claims of the present application. As

such, even the proposed combination of Korobkin and Brokenshire not teach or suggest this claim limitation.

Moreover, there is no suggestion or motivation <u>within</u> Korobkin or in Brokenshire, either alone or in combination, for one skilled in the art to combine discrete elements from Korobkin and then <u>seek out</u> still others as required by Claims 1, as described more fully below with regard to motivation to combine.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination. Further, there is a lack of proper motivation to combine these references, as discussed below. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

<u>Claim 2</u>

Claim 2 requires, among other limitations, that "the shapes are sorted into a substantially back-to-front order".

As Claim 2 depends from Claim 1, the arguments above with regard to Claim 1 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

<u>Claim 4</u>

Claim 4 requires, among other limitations, "traversing the binary-space-partition tree".

As Claim 4 depends from Claim 1, the arguments above with regard to Claim 1 apply here

as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 5

Claim 5 requires, among other limitations, that "the shapes are triangles".

As Claim 5 depends from Claim 1, the arguments above with regard to Claim 1 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 8

Independent Claim 8 requires:

A method for graphics processing, comprising:

analyzing shapes in a graphic object;

creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

performing a partition plane selection for each additional node,

classifying the shapes at the additional node according to the partition plane selection; and

creating child nodes according to the shape classification.

Examiner Nguyen <u>does not even address the limitation concerning</u> each node of a BSP tree associated with up to a predetermined number of at least one shape, as required by claim 8. As such,

Examiner Nguyen has failed to make even a *prima facie* obviousness rejection. The lack of any such teaching in any combination of the cited reference is discussed above with relation to claim 1.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination. Further, there is a lack of proper motivation to combine these references, as discussed below. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

Claim 9

Claim 9 requires, among other limitations, that "each node represents a set of elements located in a 3-dimensional spatial region".

As Claim 9 depends from Claim 8, the arguments above with regard to Claim 8 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 10

Claim 10 requires, among other limitations, that "the shapes are triangles".

As Claim 10 depends from Claim 8, the arguments above with regard to Claim 8 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 11

Claim 11 requires:

A data processing system having at least a processor and accessible memory, comprising:

means for receiving node and view data for a graphic object;

means for building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;

means for sorting shapes at each leaf of the binary-space-partition tree; and

means for outputting the sorted shapes.

Examiner Nguyen concedes that Korobkin fails to teach a binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf, as required by claim 11.

Examiner Nguyen instead cites to Brokenshire solely for allegedly teaching at least one shape associated with each leaf. Brokenshire, however, teaches a narrowly directed system for reducing the subspace boundary in a binary space partitioning tree. Examiner Nguyen refers to Brokenshire at col. 6, line 1 - col. 7, line7. For the convenience of the Board, this passage was reproduced in its entirety above with relation to claim 1.

As can be seen, although Brokenshire teaches subplanes within planes, Brokenshire fails to teach or disclose a binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf, as required by independent claims of the present application. As such, even the proposed combination of Korobkin and Brokenshire not teach or suggest this claim limitation.

Moreover, there is no suggestion or motivation within Korobkin or in Brokenshire, either alone or in combination, for one skilled in the art to combine discrete elements from Korobkin and then seek out still others as required by Claims 1, as described more fully below with regard to motivation to combine.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination. Further, there is a lack of proper motivation to combine these references, as discussed below. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

Claim 12

Claim 12 requires, among other limitations, that "the shapes are sorted into a substantially back-to-front order".

As Claim 12 depends from Claim 11, the arguments above with regard to Claim 11 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 14

Claim 14 requires, among other limitations, "traversing the binary-space-partition tree".

As Claim 14 depends from Claim 11, the arguments above with regard to Claim 11 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 15

Claim 15 requires, among other limitations, that "the shapes are triangles".

As Claim 15 depends from Claim 11, the arguments above with regard to Claim 11 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 18

Independent Claim 18 requires:

A data processing system having at least a processor and accessible memory, comprising:

means for analyzing shapes in a graphic object;

means for creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

means for performing a partition plane selection for each additional node,

means for classifying the shapes at the additional node according to the partition plane selection; and

means for creating child nodes according to the shape classification.

Examiner Nguyen <u>does not even address the limitation concerning</u> each node of a BSP tree associated with up to a predetermined number of at least one shape, as required by claim 18. As such, Examiner Nguyen has <u>failed to make even a *prima facie* obviousness rejection</u>. The lack of

any such teaching in any combination of the cited reference is discussed above with relation to claim 1.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination. Further, there is a lack of proper motivation to combine these references, as discussed below. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

Claim 19

Claim 19 requires, among other limitations, that "each node represents a set of elements located in a 3-dimensional spatial region".

As Claim 19 depends from Claim 18, the arguments above with regard to Claim 18 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 20

Claim 20 requires, among other limitations, that "the shapes are triangles".

As Claim 10 depends from Claim 18, the arguments above with regard to Claim 18 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 21

Claim 21 requires:

A computer program product tangibly embodied in a machine-readable medium, comprising:

instructions for receiving node and view data for a graphic object;

instructions for building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;

instructions for sorting shapes at each leaf of the binary-space-partition tree; and

instructions for outputting the sorted shapes.

Examiner Nguyen concedes that Korobkin fails to teach a binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf, as required by claim 21.

Examiner Nguyen instead cites to Brokenshire solely for allegedly teaching at least one shape associated with each leaf. Brokenshire, however, teaches a narrowly directed system for reducing the subspace boundary in a binary space partitioning tree. Examiner Nguyen refers to Brokenshire at col. 6, line 1 - col. 7, line7. For the convenience of the Board, this passage was reproduced in its entirety above with relation to claim 1.

As can be seen, although Brokenshire teaches subplanes within planes, Brokenshire fails to teach or disclose a binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf, as required by independent claims of the present application. As such, even the proposed combination of Korobkin and Brokenshire not teach or suggest this claim limitation.

Moreover, there is no suggestion or motivation within Korobkin or in Brokenshire, either alone or in combination, for one skilled in the art to combine discrete elements from Korobkin and then seek out still others as required by Claims 1, as described more fully below with regard to motivation to combine.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination. Further, there is a lack of proper motivation to combine these references, as discussed below. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

Claim 22

Claim 22 requires, among other limitations, that "the shapes are sorted into a substantially back-to-front order".

As Claim 22 depends from Claim 21, the arguments above with regard to Claim 21 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 24

Claim 24 requires, among other limitations, "traversing the binary-space-partition tree".

As Claim 24 depends from Claim 21, the arguments above with regard to Claim 21 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 25

Claim 25 requires, among other limitations, that "the shapes are triangles".

As Claim 25 depends from Claim 21, the arguments above with regard to Claim 21 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 28

Independent Claim 28 requires:

A computer program product tangibly embodied in a machine-readable medium, comprising:

instructions for analyzing shapes in a graphic object;

instructions for creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

instructions for performing a partition plane selection for each additional node,

instructions for classifying the shapes at the additional node according to the partition plane selection; and

instructions for creating child nodes according to the shape classification.

Examiner Nguyen <u>does not even address the limitation concerning</u> each node of a BSP tree associated with up to a predetermined number of at least one shape, as required by claim 28. As such, Examiner Nguyen has <u>failed to make even a *prima facie* obviousness rejection</u>. The lack of

any such teaching in any combination of the cited reference is discussed above with relation to claim 1.

It is clear, then, that no combination of Korobkin, Brokenshire, and Vlasic (or any other cited art), can meet the limitations of this claim. This claim is simply not taught or suggested by any art of record, alone or in combination. Further, there is a lack of proper motivation to combine these references, as discussed below. As such, the rejection should be reversed, and this claim and all its dependent claims should be allowed over the art of record.

Claim 29

Claim 29 requires, among other limitations, that "each node represents a set of elements located in a 3-dimensional spatial region".

As Claim 29 depends from Claim 28, the arguments above with regard to Claim 28 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 30

Claim 30 requires, among other limitations, that "the shapes are triangles".

As Claim 30 depends from Claim 28, the arguments above with regard to Claim 28 apply here as well, and are incorporated herein by reference. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Ground of Rejection 2: Claims 3, 6, 7, 13, 16, 17, 23, 26, and 27 were rejected as obvious over U.S. Patent No. 6,912,293 to Korobkin ("Korobkin") in view of U.S. Patent No. 6,624,810 to Brokenshire et al. ("Brokenshire") and further in view of U.S. Patent Application Publication No. 2004/0114794 to Vlasic et al. ("Vlasic").

Claim 3

Claim 3 requires, among other limitations, "caching the shape data".

As Claim 3 depends from Claim 1, the arguments above with regard to Claim 1 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. Vlasic teaches, at the cited paragraphs:

> [0050] A nave implementation of our method would not achieve real-time performance. We achieve much higher frame rates by compressing 150 textures, reducing per-frame CPU computation through caching and efficient data structures, as well as by exploiting modem graphics hardware capabilities.
>
> [0055] Although projective texture mapping runs efficiently in conventional graphics hardware, caching the projective texture coordinates can decrease processing time. Therefore, we predetermine per-vertex texture coordinates for all visible views, and store the textures within the mesh data. During rendering, each triangle is textured with the set of closest views of its three vertices.
>
> [0056] For each vertex, we define up to nine textures, nine pairs of texture coordinates, and nine blending weights. Textures and coordinates are retrieved from the set of closest views and the cached

> coordinate data, respectively. Texture weights, on the other hand, are combined as shown in FIG. 3.

This appears to generally teach the caching of texture data, unlike the claim limitation.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 6

Claim 6 requires, among other limitations, that "a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf".

As Claim 6 depends from Claim 1, the arguments above with regard to Claim 1 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. Vlasic teaches, at the cited paragraph:

> [0074] For the ULR embodiment, cosines are determined for each image. Although Delaunay triangulations are no longer needed as in the VDTM embodiment, visibility information is still useful. Therefore, per-vertex visible views are determined and stored during a preprocessing step. However, the visibility information does not need to be used for objects with relatively small occlusions. All camera images can be considered for blending at each vertex. This results in a high quality rendering. Other aspects of this embodiment, i.e., BSP tree construction and traversal, resizing textures, caching texture coordinates, combining weights, hardware shaders, and rendering, are the same as for the VDTM embodiment described above.

As can be seen, nothing in this paragraph (or any other part of Vlasic) teaches or suggests anything at all related to balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf, as required by the claim.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 7

Claim 7 requires, among other limitations, that "a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching".

As Claim 7 depends from Claim 3, the arguments above with regard to Claims 1 and 3 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. Vlasic teaches, at the cited paragraph:

> [0059] To alleviate this problem, multiple BSP trees of the same mesh are constructed by random insertion of triangles, and only the smallest tree is retained. Alternatively, as described below, a depth peeling process could be used for depth sorting of the triangle primitives, and the blending weights could then only be performed once, making the method faster. However, conventional graphics hardware is not capable of blending nine textures and computing depth layers without significant degradation in performance. Consequently, we use BSP trees for the VDTM and ULR embodiments.
>
> [0060] Opacity light fields according to the invention affects the rendering performance in another way. Each triangle must be rendered in a single pass. If there were more than one pass per

triangle, then the alpha blending is incorrect.

As can be seen, nothing in this paragraph (or any other part of Vlasic) teaches or suggests anything at all related to balancing resource usage against accuracy in the resolution of the caching.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 13

Claim 13 requires, among other limitations, "caching the shape data".

As Claim 3 depends from Claim 11, the arguments above with regard to Claim 11 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. The paragraphs of Vlasic relied upon by Examiner Nguyen were reproduced above, and appears to generally teach the caching of texture data, unlike the claim limitation.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 16

Claim 16 requires, among other limitations, that "a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf".

As Claim 16 depends from Claim 11, the arguments above with regard to Claim 11 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. The paragraph of Vlasic relied upon by Examiner Nguyen was reproduced above. Nothing in that paragraph (or any other part of Vlasic) teaches or suggests anything at all related to balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf, as required by the claim.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 17

Claim 17 requires, among other limitations, that "a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching".

As Claim 17 depends from Claim 13, the arguments above with regard to Claims 11 and 13 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. The paragraph of Vlasic relied upon by

Examiner Nguyen was reproduced above. Nothing in those paragraphs (or any other part of Vlasic) teaches or suggests anything at all related to balancing resource usage against accuracy in the resolution of the caching.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 23

Claim 23 requires, among other limitations, "caching the shape data".

As Claim 23 depends from Claim 21, the arguments above with regard to Claim 21 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. The paragraphs of Vlasic relied upon by Examiner Nguyen were reproduced above, and appears to generally teach the caching of texture data, unlike the claim limitation.

This appears to generally teach the caching of texture data, unlike the claim limitation.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 26

Claim 26 requires, among other limitations, that "a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf".

As Claim 26 depends from Claim 21, the arguments above with regard to Claim 21 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. The paragraph of Vlasic relied upon by Examiner Nguyen was reproduced above. Nothing in that paragraph (or any other part of Vlasic) teaches or suggests anything at all related to balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf, as required by the claim.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Claim 27

Claim 27 requires, among other limitations, that "a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching".

As Claim 27 depends from Claim 23, the arguments above with regard to Claims 21 and 23 apply here as well, and are incorporated herein by reference, and applicant notes that the teachings missing from the proposed Korobkin/Brokenshire combination are also not taught by a Korobkin/Brokenshire/Vlasic combination. Nothing in the art of record teaches or suggests this feature in combination with the other features of the parent claim, as described above.

Examiner Nguyen concedes that this feature is not taught or suggested by the proposed Korobkin/Brokenshire combination, and looks to Vlasic. The paragraph of Vlasic relied upon by

Examiner Nguyen was reproduced above. Nothing in those paragraphs (or any other part of Vlasic) teaches or suggests anything at all related to balancing resource usage against accuracy in the resolution of the caching.

The rejection of this claim should be reversed, and it should be allowed over all art of record.

Therefore, all claims should be allowed over the combination of Korobkin, Brokenshire, and Vlasic, and Examiner Nguyen's obviousness rejections should be reversed.

<u>Motivation to Combine or Modify</u>[2]

Examiner Nguyen makes a variety of statements as alleged "motivations" to combine Korobkin with Brokenshire and Vlasic, but these "motivations" are not taught or suggested by the art of record, and are not specific to the modifications and combinations she purports to support with them. A proper obviousness rejection must include a proper motivation: "In holding an invention obvious in view of a combination of references, there must be some suggestion, motivation, or teaching in the prior art that would have led a person of ordinary skill in the art to select the references and <u>combine them in the way that would produce the claimed invention</u>." (*Karsten Mfg. Corp. v. Cleveland Golf Co.*, 242 F.3d 1376, 1385 (Fed. Cir. 2001) emphasis added). "When the references are in the same field as that of the applicant's invention, knowledge thereof is presumed. However, the test of whether it would have been obvious to select specific teachings and <u>combine them as did the applicant</u> must still be met by identification of some suggestion, teaching, or motivation in the prior art, arising from what the prior art would have taught a person of ordinary

---

[2]Where an obviousness rejection is based on a combination of references, the Examiner must show that one of ordinary skill would have been motivated to combine those references. *See In re Nilssen*, 7 USPQ2d 1500 (Fed.Cir. 1988); *Panduit Corp. v. Dennison Mfg. Co.*, 1 USPQ2d 1593, 1597 (Fed.Cir. 1987); *ACS Hospital Systems v. Montefiore Hospital*, 220 USPQ 929 (Fed.Cir. 1984).

"When prior art references require selective combination ... to render obvious a subsequent invention, there must be some reason for the combination other than the hindsight gained from the invention itself.... Something in the prior art as a whole must suggest the desirability, and thus the obviousness, of making the combination." *Uniroyal, Inc. v. Rudkin-Wiley Corp.*, 5 USPQ2d 1434, 1438 (Fed.Cir. 1988), *quoting Interconnect Planning Corp. v. Feil*, 227 USPQ 543 (Fed.Cir. 1985), *and Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick*, 221 USPQ 481 (Fed.Cir. 1984).

"While *[a reference]* may be capable of being modified to run the way *[the applicant's]* apparatus is Claimed, there must be a suggestion or motivation in the reference to do so. See In re Gordon, 733 F.2d 900, 902, 221 USPQ 1125, 1127 (Fed. Cir. 1984) ("The mere fact that the prior art could be so modified would not have made the modification obvious unless the prior art suggested the desirability of the modification."). *In re Mills,* 916 F.2d 680, 16 U.S.P.Q.2d 1430 (Fed.Cir. 1990).

skill in the field of the invention." (*In re Dance*, 160 F.3d 1339, 1343 (Fed. Cir. 1998), emphasis added).

In short, there appears to be no teaching or suggestion in the art to motivate one to modify Korobkin with the teachings of Brokenshire and Vlasic.

The Examiner states that a motivation for combining Korobkin with the teachings of Brokenshire is "to accomplish a global visibility sort of the input database of Korobkin system, because once the space has been subdivided into a predetermined number of level of [*sic*] subspaces, the bounding volume of each space or subspace is recomputed such that the bounding volume just contains the object that fits into that level of the BSP tree (col. 7, lines 3-14)".

This reasoning is flawed – Korobkin does not teach anything related to the BSP tree having up to a predetermined number of at least one shape associated with each leaf, as claimed, and this is the teaching that the Examiner attempts to drawn from Brokenshire (though it is not taught by Brokenshire, either). As can be seen, Examiner Nguyen's suggested "motivation" does not motivate one to make the proposed combination/modification at all.

The Examiner makes a similar allegation of motivation to combine Vlasic with Korobkin and Brokenshire: "to accomplish a global visibility sort of the input database, because it would provide high quality and high performance rendering (paragraph 0076)." Even if the Vlasic did teach the claimed balancing techniques as alleged by the Examiner – which it does not – this motivation has nothing at all to do with that, and clearly is not <u>specific</u> to the proposed modification or combination, as required.

As such, there is no properly motivation combination stated that would support the Examiner's rejections.

Grouping of Claims

The claims on appeal do not stand or fall together, as may be seen from the arguments set forth below. Each claim has been argued separately under a separate subheading, and each claim should be considered separately. While the applicant recognizes that a formal statement regarding the grouping of claims is no longer required, each claim should be considered separately; or at the very least each claim which is argued separately in the preceding sections of this brief should be considered separately. Argument: The fact that the claims use different formulations (as detailed above) and/or have been argued separately, shows that, if their patentability is not considered separately, any adverse decision would show that the limitations of some claims had been unfairly ignored.

REQUESTED RELIEF

The Board is respectfully requested to reverse the outstanding rejections and return this application to the Examiner for allowance.

Respectfully submitted,
MUNCK BUTRUS, P.C.

Date: 1/9/7

Matthew S. Anderson
Registration No. 39,093

P.O. Drawer 800889
Dallas, Texas 75380
Phone: (972) 628-3600
Fax: (972) 628-3616
E-mail: manderson@munckbutrus.com

Attorney for Applicant

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

APPENDIX A -

Claims Appendix

1. (Previously Presented) A method for graphics processing, comprising:
receiving node and view data for a graphic object;
building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;
sorting shapes at each leaf of the binary-space-partition tree; and
outputting the sorted shapes.

2. (Original) The method of claim 1, wherein the shapes are sorted into a substantially back-to-front order.

3. (Original) The method of claim 1, further comprising caching the shape data.

4. (Original) The method of claim 1, further comprising traversing the binary-space-partition tree.

5. (Original) The method of claim 1, wherein the shapes are triangles.

6. (Original) The method of claim 1, wherein a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf.

7. (Original) The method of claim 3, wherein a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching.

8. (Previously Presented) A method for graphics processing, comprising:

analyzing shapes in a graphic object;

creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

performing a partition plane selection for each additional node,

classifying the shapes at the additional node according to the partition plane selection; and

creating child nodes according to the shape classification.

9. (Original) The method of claim 8, wherein each node represents a set of elements located in a 3-dimensional spatial region.

10. (Original) The method of claim 8, wherein the shapes are triangles.

11. (Previously Presented) A data processing system having at least a processor and accessible memory, comprising:

means for receiving node and view data for a graphic object;

means for building a binary-space-partition tree corresponding to the graphic object, the

binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;

means for sorting shapes at each leaf of the binary-space-partition tree; and

means for outputting the sorted shapes.

12. (Original) The data processing system of claim 11, wherein the shapes are sorted into a substantially back-to-front order.

13. (Original) The data processing system of claim 11, further comprising means for caching the shape data.

14. (Original) The data processing system of claim 11, further comprising means for traversing the binary-space-partition tree.

15. (Original) The data processing system of claim 11, wherein the shapes are triangles.

16. (Original) The data processing system of claim 11, wherein a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf.

17. (Original) The data processing system of claim 13, wherein a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching.

18. (Previously Presented) A data processing system having at least a processor and accessible memory, comprising:

means for analyzing shapes in a graphic object;

means for creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

means for performing a partition plane selection for each additional node,

means for classifying the shapes at the additional node according to the partition plane selection; and

means for creating child nodes according to the shape classification.

19. (Original) The data processing system of claim 18, wherein each node represents a set of elements located in a 3-dimensional spatial region.

20. (Original) The data processing system of claim 18, wherein the shapes are triangles.

21. (Previously Presented) A computer program product tangibly embodied in a machine-readable medium, comprising:

instructions for receiving node and view data for a graphic object;

instructions for building a binary-space-partition tree corresponding to the graphic object, the binary-space-partition tree having up to a predetermined number of at least one shape associated with each leaf;

instructions for sorting shapes at each leaf of the binary-space-partition tree; and

instructions for outputting the sorted shapes.

22. (Original) The computer program product of claim 21, wherein the shapes are sorted into a substantially back-to-front order.

23. (Original) The computer program product of claim 21, further comprising instructions for caching the shape data.

24. (Previously Presented) The computer program product of claim 21, further comprising

instructions for traversing the binary-space-partition tree.

25. (Original) The computer program product of claim 21, wherein the shapes are triangles.

26. (Original) The computer program product of claim 21, wherein a configuration component is used, the configuration component balancing the resolution of the binary-space-partition tree against the sorting shapes at each leaf.

27. (Original) The computer program product of claim 23, wherein a configuration component is used, the configuration component balancing resource usage against accuracy in the resolution of the caching.

28. (Previously Presented) A computer program product tangibly embodied in a machine-readable medium, comprising:

instructions for analyzing shapes in a graphic object;

instructions for creating a root node and a list of additional nodes for a binary-space-partition tree, each node associated with up to a predetermined number of at least one shape;

instructions for performing a partition plane selection for each additional node,

instructions for classifying the shapes at the additional node according to the partition plane selection; and

instructions for creating child nodes according to the shape classification.

29. (Original) The computer program product of claim 28, wherein each node represents a set of elements located in a 3-dimensional spatial region.

30. (Original) The computer program product of claim 28, wherein the shapes are triangles.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

APPENDIX B

Copy of Formal Drawings

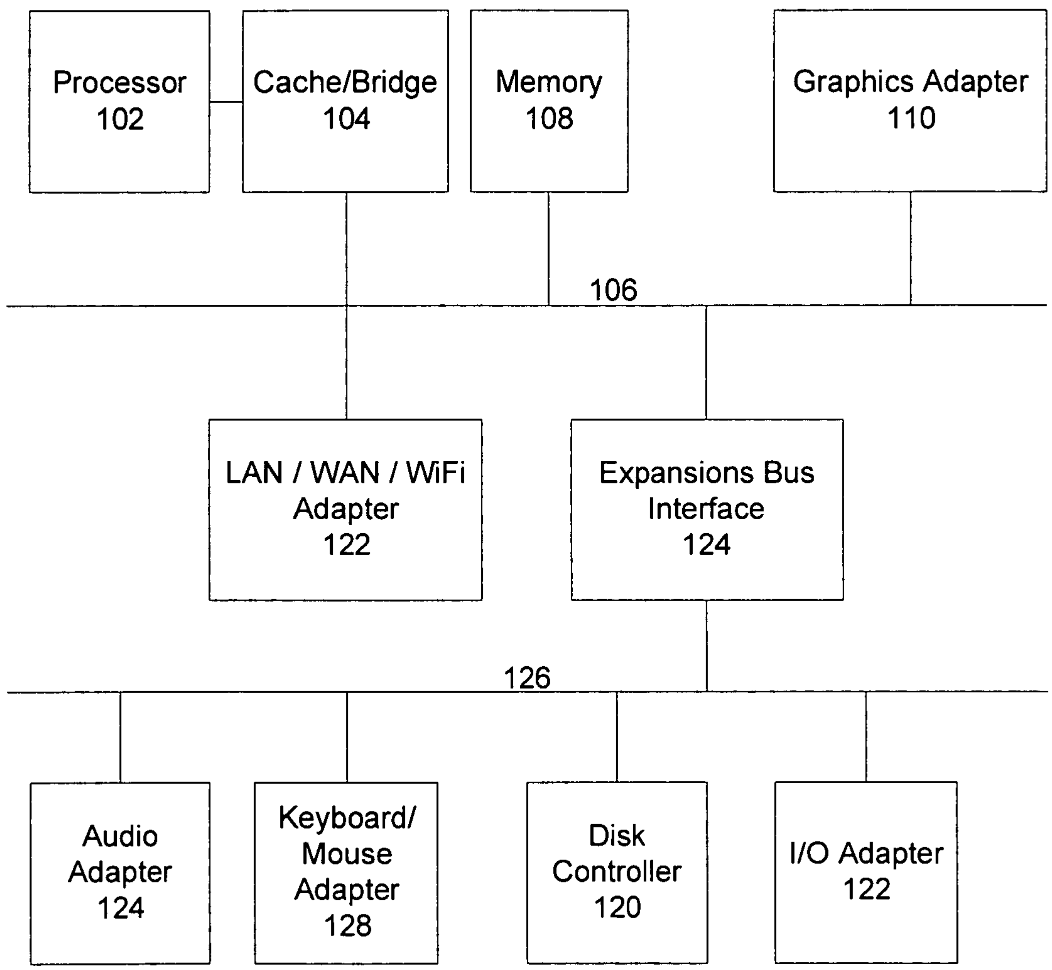

Processor
102
Cache/Bridge
104
Memory
108
Graphics Adapter
110
106
LAN / WAN / WiFi
Adapter
122
Expansions Bus
Interface
124
126
Audio
Adapter
124
Keyboard/
Mouse
Adapter
128
Disk
Controller
120
I/O Adapter
122

Figure 1

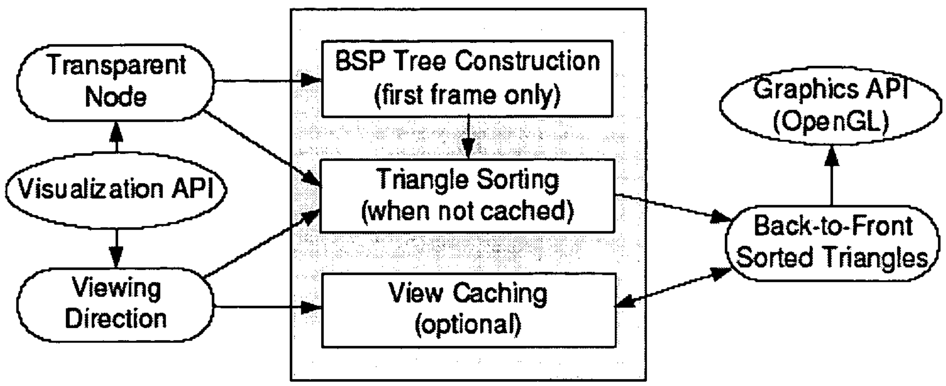

Transparent Node
Visualization API
Viewing Direction
BSP Tree Construction (first frame only)
Triangle Sorting (when not cached)
View Caching (optional)
Graphics API (OpenGL)
Back-to-Front Sorted Triangles

Figure 2

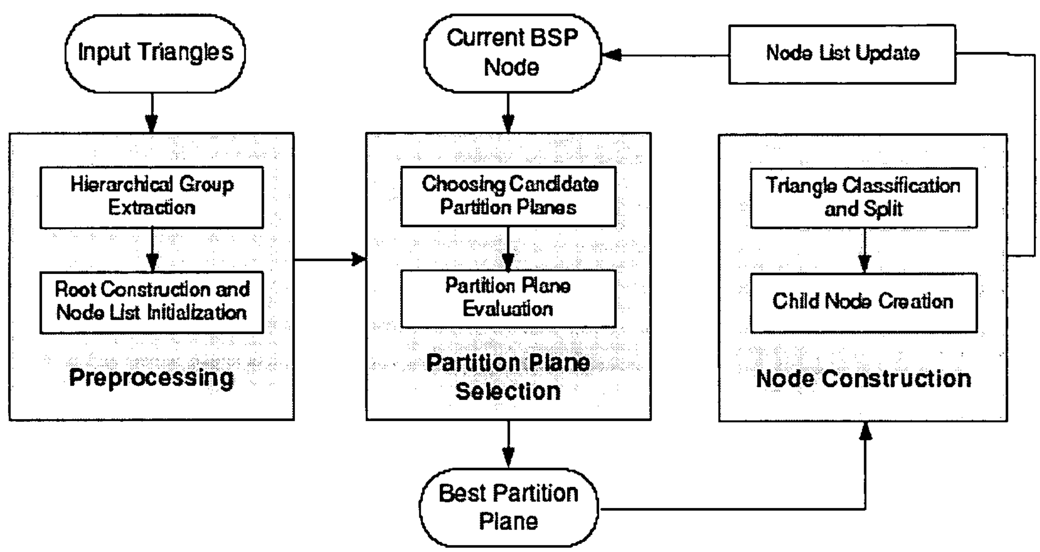

Input Triangles
Hierarchical Group Extraction
Root Construction and Node List Initialization
Preprocessing
Current BSP Node
Choosing Candidate Partition Planes
Partition Plane Evaluation
Partition Plane Selection
Best Partition Plane
Node List Update
Triangle Classification and Split
Child Node Creation
Node Construction

Figure 3

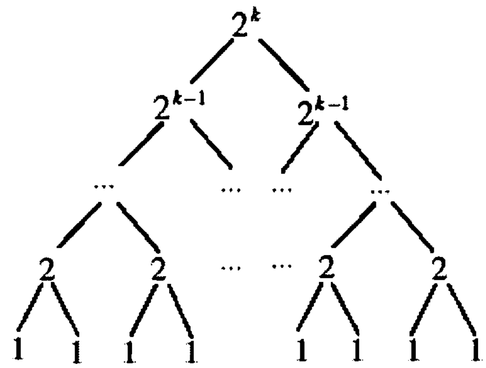
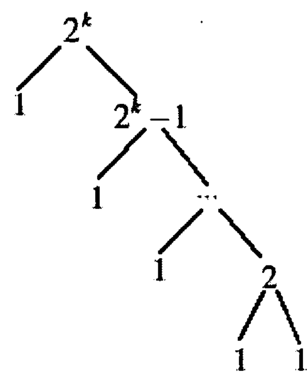


Figure 4A

Figure 4B



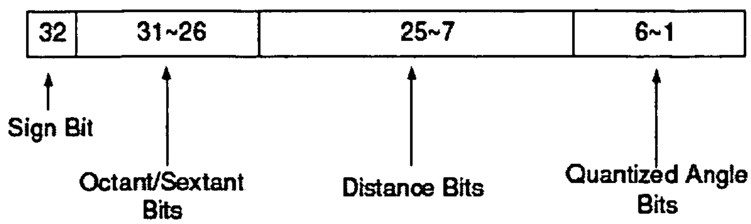


Figure 5

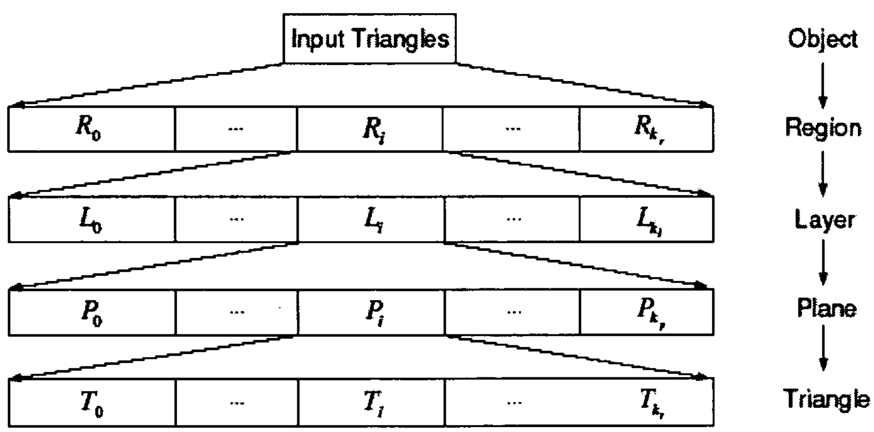


Figure 6

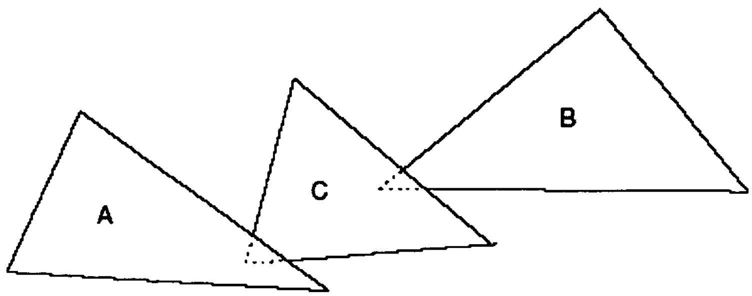


Figure 7

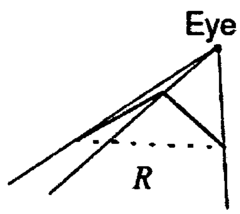


Figure 8A

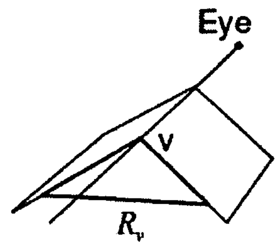


Figure 8B

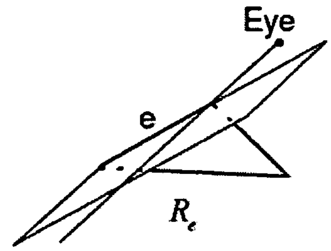


Figure 8C

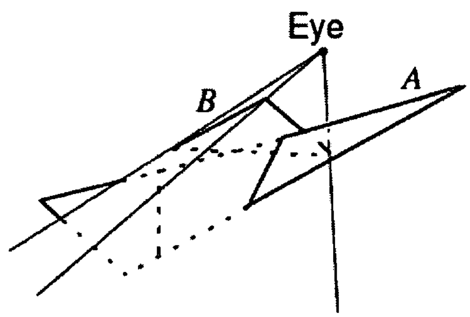


Figure 9

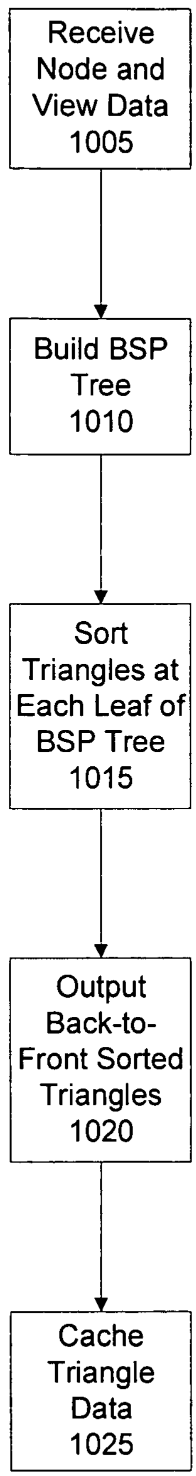

Receive Node and View Data 1005
Build BSP Tree 1010
Sort Triangles at Each Leaf of BSP Tree 1015
Output Back-to-Front Sorted Triangles 1020
Cache Triangle Data 1025

Figure 10

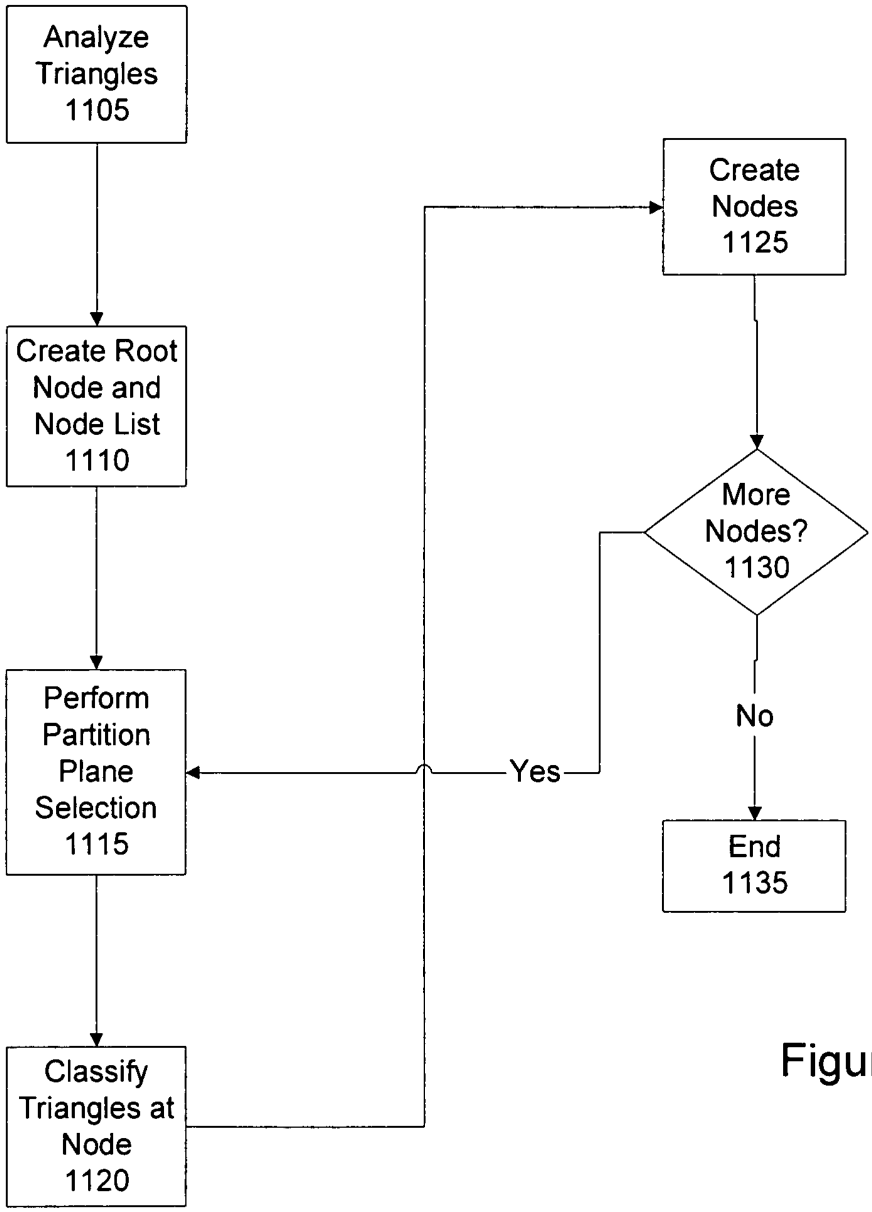

Analyze Triangles 1105
Create Root Node and Node List 1110
Perform Partition Plane Selection 1115
Classify Triangles at Node 1120
Create Nodes 1125
More Nodes? 1130
Yes
No
End 1135

Figure 11

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

APPENDIX C
Evidence Appendix

None.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Jianbing Huang, et al.
Serial No.:	10/731,281
Filed:	December 9, 2003
For:	SYSTEM AND METHOD FOR TRANSPARENCY RENDERING
Group No.:	2628
Examiner:	Kimbinh T. Nguyen

APPENDIX D

Related Proceedings Appendix

None.